An ant colony approach to redundancy optimization for multi-state systems

Mustapha Nourelfath and Nabil Nahas
CENTOR Network Organization Technology Research Center
University of Quebec in Abitibi-Temiscamingue (Canada)
mustapha.nourelfath@uqat.ca

Abstract

An ant colony meta-heuristic optimization method is used to solve the redundancy optimization problem for multi-state systems. A universal generating function technique is applied to evaluate system availability. The ant colony approach developed and tested in this paper has the advantage to allow elements with different parameters to be allocated in parallel.

1. Introduction

The redundancy optimization problem is a well known combinatorial optimization problem. In (Levitin *et al.*, 1997), the authors applied a genetic algorithm to solve the problem. This paper uses an ant colony meta-heuristic optimization method. The system and its components have a range of performance levels from perfect functioning to complete failure. Redundant elements are included in order to achieve a desirable level of availability. The elements of the system are characterized by their cost, performance and availability. These elements are chosen from a list of products available on the market. The proposed meta-heuristic determines the minimal cost system configuration under availability constraints. During the optimization process, artificial ants will have to evaluate the availability of a given selected structure of the series-parallel system. To do this, a fast procedure of availability estimation is developed. This procedure is based on a modern mathematical technique: the *z*-transform or universal moment generating function which was introduced in (Ushakov, 1986). It was proven to be very effective for high dimension combinatorial problems. A good and extensive recent review of the literature can be found for example in (Ushakov, Levitin and Lisnianski, 2002) or (Lisnianski and Levitin, 2003). The developed method allows the availability function of reparable series-parallel MSS to be obtained using a straightforward numerical procedure. The paper is organized as follows. The ant system approach is summarized in section 2. In section 3, the proposed heuristic is presented

2. The ant system approach

Many researchers have shown that insect colonies behaviour can be seen as a natural model of collective problem solving. The analogy between the way ants look for food and combinatorial optimization problems has given rise to a new computational paradigm, which is called ant colony meta-heuristic. Ants lay down in some quantity an aromatic substance, known as *pheromone*, in their way to food. The pheromone quantity depends on the length of the path and the quality of the discovered food source. An ant chooses a specific path in correlation with the intensity of the pheromone. The pheromone trail evaporates over time if no more pheromone is laid down. Other ants can observe the pheromone trail and are attracted to follow it. Thus, the path will be marked again and will therefore attract more ants. The pheromone trail on paths leading to rich food sources close to the nest will be more frequented and will therefore grow faster. In that way, the best solution has more intensive pheromone and higher probability to be chosen. The described behaviour of real ant colonies can be used to solve combinatorial optimization problems by simulation: artificial ants searching the solution space simulate real ants searching their environment. The objective values correspond to the quality of the food sources. The ant system approach associates pheromone trails to features of the solutions of a combinatorial problem, which can be seen as a kind of adaptive memory of the previous solutions. In order to demonstrate the ant system approach, Dorigo *et al.* (1996) apply it to the classical traveling salesman problem, asymmetric traveling salesman problem, quadratic assignment problem, and job-shop scheduling. Ant system shows very good results in each applied area. The ant system has also been applied with success to other combinatorial optimization problems. The ant system method has not yet been used in the redundancy optimization of multi-state systems.

3. A solution approach to redundancy optimization for multi-state systems

3.1 Problem formulation

Let consider a series-parallel system containing n components C_i ($i = 1, 2,\ldots, n$) in series. Every component C_i contains a number of different elements connected in parallel. For each component i, there are a number of element versions available in the market. For any given system component, different versions and number of elements may be chosen. For each component i, elements are characterized according to their version v by their cost (C_{iv}), availability (A_{iv}) and performance (Σ_{iv}). The structure of system component i can be defined by the numbers of parallel elements (of each version) k_{iv} for $1 \le v \le V_i$, where V_i is a number of versions available for element of type i. The entire system structure is defined by the vectors $\boldsymbol{k_i} = \{k_{iv_i}\}$ $(1 \le i \le n, 1 \le v \le V_i)$. For a given set of vectors $\boldsymbol{k_1}, \boldsymbol{k_2},\ldots, \boldsymbol{k_n}$ the total cost of the system can be calculated as:

$$C = \sum_{i=1}^{n} \sum_{v=1}^{V_i} k_{iv} C_{iv} \tag{1}$$

The unfamiliar reader is referred to (Ushakov, 1986) or (Lisnianski and Levitin, 2003) for the universal moment generating function (UMGF) method, habitually used to estimate the availability of reparable multi-state systems. The multi-state system redundancy optimization problem can be formulated as follows: find the minimal cost system configuration $\boldsymbol{k_1}, \boldsymbol{k_2}, \ldots, \boldsymbol{k_n}$ that meets or exceeds the required availability A_0. That is,

Minimize $$C = \sum_{i=1}^{n} \sum_{v=1}^{V_i} k_{iv} C_{iv} \tag{2}$$

Subject to $$A(\boldsymbol{k_1}, \boldsymbol{k_2}, \ldots, \boldsymbol{k_n}, \boldsymbol{D}, \boldsymbol{T}) \ge A_0 \tag{3}$$

3.2 Solution method

To apply the ACO meta-heuristic to a combinatorial optimization problem, it is convenient to represent the problem by a graph $G = (\varsigma, \Lambda)$, where ς are the nodes and Λ is the set of edges. To represent our ROP as such a graph, the set of nodes ς is given by components and elements, and edges connect each component to its available elements. Some nodes are added to represent positions where additional component was not used. As in (Liang and Smith, 2001), these nodes are called blanks nodes and have attributes of zero. The obtained graph is partially connected. Ants cooperate by using indirect form of communication mediated by pheromone they deposit on the edges of the graph G while building solutions. Informally, our algorithm works as follows: m ants are initially positioned on a node representing a component. Each ant represents one possible structure of the entire system. This structure is represented by K_i elements in parallel for n different components. The K_i elements can be chosen in any combination from the V_i available type of elements. Each ant builds a feasible solution (called a tour) to the ROP problem by repeatedly applying a stochastic greedy rule, i.e., the *state transition rule*. While constructing its solution, an ant also modifies the amount of pheromone on the visited edges by applying the *local updating rule*. Once all ants have terminated their tour, the amount of pheromone on edges is modified again (by applying the *global updating rule*). Ants are guided, in building their tours, by both heuristic information (they prefer to choose "less expansive" edges), and by pheromone information. Naturally, an edge with a high amount of pheromone is a very desirable choice. The pheromone updating rules are designed so that they tend to give more pheromone to edges which should be visited by ants. At each step of the construction process, ants use problem-specific heuristic information (denoted by η_{ij}) and pheromone trails (denoted by τ_{ij}) to select K_i elements in each component. An ant positioned on node i (representing a component C_i) chooses the element j by applying the rule given by:

$$j = \begin{cases} arg \max_{m \in AC_i} \left\{ (\tau_{im})^{\alpha} (\eta_{im})^{\beta} \right\} & if \quad q \leq q_o \\ J & otherwise \end{cases} \tag{4}$$

and J is a random variable selected according to the probability distribution given by:

$$p_{ij} = \begin{cases} \dfrac{(\tau_{ij})^{\alpha} (\eta_{ij})^{\beta}}{\sum_{m \in AC_i} (\tau_{im})^{\alpha} (\eta_{im})^{\beta}} & if \quad j \in AC_i \\ 0 & otherwise \end{cases} \tag{5}$$

where α and β are parameters that control the relative weight of the pheromone (τ_{ij}) and the local heuristic (η_{ij}), respectively; AC_i is the set of available element choices for component i; q is a random number uniformly distributed in [0 .. 1]; and q_0 is a parameter ($0 \leq q_0 \leq 1$).

The parameter q_0 determines the relative importance of exploitation versus exploration: every time an ant in node i has to choose an element j, it samples a random number $0 \leq q \leq 1$. If $q \leq q_o$ then the best edge, according to equation (4), is chosen (exploitation), otherwise an edge is chosen according to equation (5) (biased exploration). The state transition rule resulting from equations (4) and (5) is a *pseudo-random-proportional rule*.

The heuristic information used is $\eta_{ij} = 1/(1+c_{ij})$ where c_{ij} represents the associated cost of element j for component i. In equation (5) we multiply the pheromone on edges by the corresponding heuristic value. In this way we favour the choice of edges which are weighted with smaller costs and which have a greater amount of pheromone. That is, elements with smaller cost have greater probability to be selected.

Once all ants have built a complete system, pheromone trails are updated. Only the globally best ant (i.e., the ant which constructed the best design solution from the beginning of the trial) is allowed to deposit pheromone. A quantity of pheromone $\Delta\tau_{ij}$ is deposited a on each edge that the best ant has used. The quantity $\Delta\tau_{ij}$ is given by $\frac{1}{TC_{best}}$, where TC_{best} is the total cost of the design feasible solution constructed by the best ant. Therefore, the global updating rule is:

$$\tau_{ij} \leftarrow (1-\rho).\tau_{ij} + \rho.\Delta\tau_{ij} \tag{6}$$

where $0 < \rho < 1$ is the pheromone decay parameter representing the evaporation of trail.
Global updating is intended to allocate a greater amount of pheromone to less expansive design solution. Equation (6) dictates that only those edges belonging to the globally best solution will receive reinforcement.
While building a solution of the ROP problem of MSS, ants choose elements by visiting edges on the graph *G*, and change their pheromone level by applying the following local updating rule:

$$\tau_{ij} \leftarrow (1-\rho).\tau_{ij} + \rho.\tau_{o} \tag{7}$$

where ρ is a coefficient such that $(1-\rho)$ represents the evaporation of trail; and τ_o is the initial value of trail intensities.
The application of the local updating rule, while edges are visited by ants, has the effect of lowering the pheromone on visited edges. The pheromone reduction is small but sufficient to lower the attractiveness (or desirability) of precedent edge. This favors the exploration of edges not yet visited, since the visited edges will be chosen with a lower probability by the other ants in the remaining steps of an iteration of the algorithm. Thus, by discouraging the next ant from choosing the same element during the same cycle, ants never converge to a common solution and premature convergence is avoided.

Note finally that numerical results of the proposed approach are omitted for lack of space and can be found in (Nourelfath *et al.*, 2003).

Acknowledgements

The authors would like to thank *NSERC*, *NATEQ* and *FUQAT* for financial support.

References

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics- Part B*, 26(1), 1-13.

Lisnianski, A., & Levitin, G. (2003). Multi-state system reliability. Assesment, optimization and applications. World scientific; 2003.

Levitin, G., Lisnianski, A., Ben-Haim, H., & Elmakis, D. (1997). Structure optimization of power system with different redundant elements. *Electric Power Systems Research*, 43(1), 19-27.

Liang, Y. C., & Smith, A. (1999). An Ant System to Redundancy Allocation. *Proceedings of the IEEE Congress on Evolutionary Computation*, 1478-1484.

Nourelfath, M., Zeblah, A., & Nahas, N. (2003). An ant colony approach to redundancy optimization for multi-state systems. *International Conference on Industrial Engineering and Production Management (IEPM'2003)*, Porto, 26-28 Mai 2003.

Ushakov, I., Levitin, G., & Lisnianski, A. (2002). Multi-state system reliability: from theory to practice. *Proc. of 3 Int. Conf. on mathematical methods in reliability, MMR 2002* (pp. 635-638). Trondheim, Norway.

Ushakov, I. (1986). Universal generating function. *Sov. J. Computing System Science*, 24(5), 118-129.